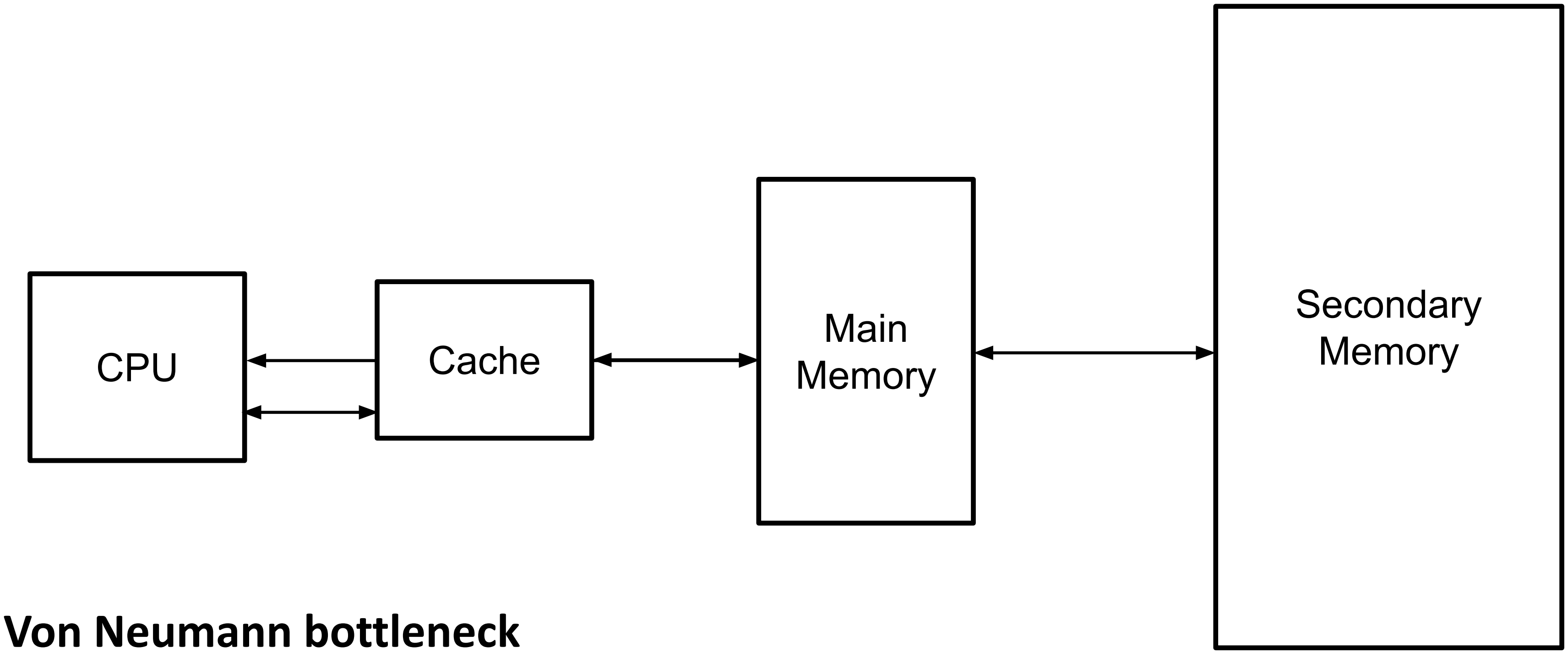


Von Neumann bottleneck

- The CPU is very much faster than main memory.
- CPU try to access data and programs from main memory. Since main memory is slower than CPU so it can not be accessed with the same speed of CPU.
- Disparity of speed between CPU and main memory minimize the performance of CPU. This is called Von Neumann bottleneck.

To overcome this situation one very fast and small memory is allowed in between the CPU and main memory path whose access time is near about CPU. This small high speed memory is called cache (SRAM).

Locality of reference:

Computer program tends to access same localized areas in a memory for a particular time period. This phenomenon is known as the property of locality of reference.

Cache follows the locality of reference principal.

Those instructions and data of the main memory are frequently referred that are kept In cache memory.

There are two types of Locality of reference

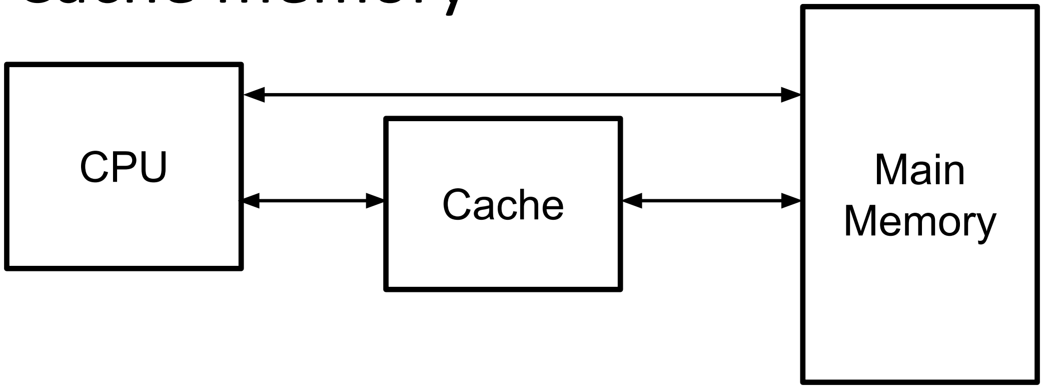
i)Temporal locality:

Recently referenced instructions are likely to be referenced again in the near future. This is called temporal locality. For example, in case of iterative loops, subroutines, a small code segment will be referenced repeatedly.

ii) Spatial locality:

This refers to the tendency for a program to access instructions whose addresses are near one another. For example, in case of arrays, memory accesses are generally consecutive addresses.

Cache memory



When the CPU refers the memory for a word. If the word found in cache it is called cache hit and if not found then cache miss.

The ratio of number of hits and the total number of CPU references to the memory is called hit ratio.

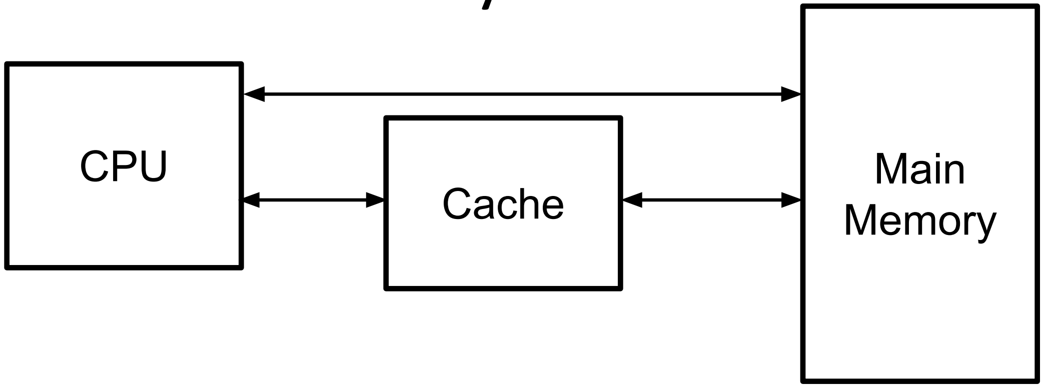
Total number of CPU references to the memory = number of hits + number of misses

Miss ratio = = 1- Hit ratio

$$\text{Hit ratio} = \frac{\text{number of hits}}{\text{number of hits+number of misses}}$$

$$\text{Miss ratio} = \frac{\text{number of misses}}{\text{number of hits+number of misses}} = 1\text{- Hit ratio}$$

Cache memory contd.



Cache access time= t_c
Main memory access time = t_m
Hit ratio is h_c
Average Memory Access Time = $h_c \times t_c + (1- h_c) \times (t_m + t_c)$

For example a computer with cache access time 100 ns , a main memory access time is 1000 ns and a hit ratio of 0.9. Calculate the Average Memory Access Time.

t_c = 100 ns
t_m = 1000 ns
h_c = 0.9
Average Memory Access Time =0.9 x 100+ (1- 0.9) x (1000 + 100) ns
= 90 +110 ns
= 200 ns

Thank You